

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平10-40035

(43)公開日 平成10年(1998) 2月13日

(51)Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F	3/12		G 0 6 F 3/12	D
				C
B 4 1 J	29/38		B 4 1 J 29/38	Z

審査請求 未請求 請求項の数6 O L (全 8 頁)

(21)出願番号 特願平8-189564

(22)出願日 平成8年(1996) 7月18日

(71)出願人 000005496

富士ゼロックス株式会社
東京都港区赤坂二丁目17番22号

(72)発明者 宇田 豊和

神奈川県川崎市高津区坂戸3丁目2番1号
K S P R & Dビジネスパークビル 富
士ゼロックス株式会社内

(72)発明者 小谷川 修

神奈川県川崎市高津区坂戸3丁目2番1号
K S P R & Dビジネスパークビル 富
士ゼロックス株式会社内

(74)代理人 弁理士 吉田 研二 (外2名)

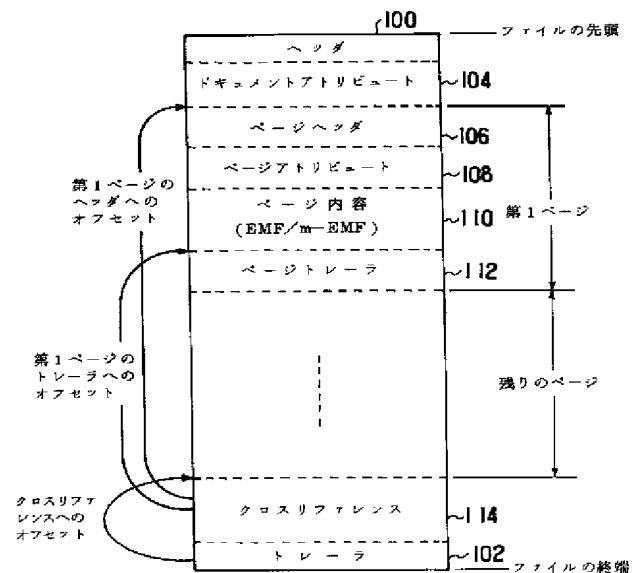
最終頁に続く

(54)【発明の名称】 印刷処理装置

(57)【要約】 (修正有)

【課題】 クライアントが、印刷ジョブを作成する場合に、その負担を軽減する。

【解決手段】 クライアントが作成した中間形態ファイル及びその区切り情報を記述した領域を含む印刷ジョブに対して、プリントサーバが印刷情報を記述した情報記述ファイルを完成する。クライアント側では、ハンドル数とレコード数にダミー値を入れたモディファイドE M F (m - E M F) を作成し、ハンドル数とレコード数は、ページトレーラ112に書き込み、印刷ジョブを作成する。この印刷ジョブを受けたプリントサーバは、ページトレーラのハンドル数とレコード数を読みだし、ページ内容110に書き戻し、E M F を完成させ、その後、印刷データにデコンポーズする。m - E M F 作成時にハンドル数とレコード数を計数し、続いてこれをトレーラに格納するので、1パス処理ができる。



【特許請求の範囲】

【請求項1】 印刷ジョブを送信するクライアントと、クライアントからの印刷ジョブを受信し、印刷ジョブに含まれる印刷情報を記述した情報記述ファイルを印刷データに展開し、当該印刷データに基づき印刷を行なうプリントサーバと、を有する印刷処理装置において、前記クライアントは、印刷データに展開される前記情報記述ファイルに対して中間の形態である中間形態ファイルを作成する中間形態ファイル作成部と、中間形態ファイルの区切り情報を記述した領域を作成する区切り領域作成部と、前記中間形態ファイルと区切り領域を含む印刷ジョブを作成する印刷ジョブ作成部と、を含み、前記プリントサーバは、前記中間形態ファイルと区切り情報に基づき前記情報記述ファイルを完成させる情報記述ファイル作成部を含む、印刷処理装置。

【請求項2】 請求項1に記載の印刷処理装置において、前記区切り情報には、前記中間形態ファイル内の描画処理時に参照される値が記載され、前記情報記述ファイル作成部において、前記描画処理時に参照される値を前記中間形態ファイルに書き込むことにより前記情報記述ファイルを完成させる、印刷処理装置。

【請求項3】 請求項1または2に記載の印刷処理装置において、印刷ジョブ作成部は、前後に前記区切り領域を付加された中間形態ファイルを複数個包含する包含ファイルを作成する包含ファイル作成部を含む、印刷処理装置。

【請求項4】 請求項3に記載の印刷処理装置において、前記包含ファイル作成部は、前記複数の中間形態ファイルの各々の格納された領域の位置を示す情報が記述される目次情報領域作成部を含む、印刷処理装置。

【請求項5】 請求項3に記載の印刷処理装置において、印刷ジョブ作成部は、先行して前記情報記述ファイルを作成する情報記述ファイル先行作成部を含み、前記区切り領域作成部は、前記先行して作成された情報記述ファイルの区切り情報も作成し、前記包含ファイル作成部は、各々前記区切り領域が付加された、前記先行して作成された情報記述ファイルと前記中間形態ファイルを、包含する包含ファイルを作成する、印刷処理装置。

【請求項6】 請求項5に記載の印刷処理装置において、前記区切り領域には、当該区切り領域に対応するファイルが、情報記述ファイルであるか中間形態ファイルであるかの情報が記述されている、印刷処理装置。

【発明の詳細な説明】**【0001】**

【発明の属する技術分野】 本発明は、印刷ジョブを送信するクライアントと、前記印刷ジョブに含まれる印刷情報を印刷データに展開し、当該印刷データに基づき印刷を行なうプリントサーバとを有する印刷処理装置に関し、特に、送受信される印刷情報を記述したファイルの形式に関する。

【0002】

【従来の技術】 クライアントからの要求に従ってプリントサーバにて印刷を行う印刷処理装置が知られている。このような装置の場合、一つのプリントサーバに対して、複数のクライアントを、離れた所に設置することも可能であり、いわゆるローカル・エリア・ネットワーク（LAN）の一部を形成する装置となっている。

【0003】 プリントサーバは、一つまたは複数のファイル形式によって記述された印刷情報をプリンタを駆動するための印刷データに変換して、プリンタにより印刷を行っている。したがって、プリントサーバに送信される情報は、このサーバにおいて印刷データに変換可能な形式で記述されている必要がある。

【0004】

【発明が解決しようとする課題】 前述のようなクライアントとプリントサーバの間で送受信されるデータのファイル形式は各種提案されているが、このファイル形式の中には、ファイル作成時のクライアントの負担が大きい形式がある。このような形式でファイルを作成する場合は、クライアントでの作業が制限されるという問題があった。

【0005】 本発明は前述の問題点を解決するためになされたものであり、印刷処理に関して、クライアント側の負担を軽減することのできる印刷処理装置を提供することを目的とする。

【0006】

【課題を解決するための手段】 前述の目的を達成するために、本発明にかかる印刷処理装置は、印刷ジョブを送信するクライアントと、クライアントからの印刷ジョブを受信し、印刷ジョブに含まれる印刷情報を記述した情報記述ファイルを印刷データに展開し、当該印刷データに基づき印刷を行なうプリントサーバと、を有する印刷処理装置であって、前記クライアントは、印刷データに展開される前記情報記述ファイルに対して中間の形態である中間形態ファイルを作成する中間形態ファイル作成部と、中間形態ファイルの区切り情報を記述した領域を作成する区切り領域作成部と、前記中間形態ファイルと区切り領域を含む印刷ジョブを作成する印刷ジョブ作成部とを含み、前記プリントサーバは、前記中間形態ファイルと区切り情報に基づき前記情報記述ファイルを完成させる情報記述ファイル作成部を含んでいる。

【0007】 この構成によれば、クライアントで最終的なファイル形式である情報記述ファイルを作成する必要がなく、中間的な形式のファイルを一旦作成し、これをプリントサーバに送出するので、クライアントの負担を低減することができる。そして、プリントサーバ側で中間的な形式のファイルを書き替えて、最終的なファイル形式を得ている。

【0008】 さらに、本発明の一つの実施の態様によれば、前記区切り情報には、前記中間形態ファイル内の描

画処理時に参照される値が記載され、前記情報記述ファイル作成部において、前記描画処理時に参照される値を前記中間形態ファイルに書き込むことにより前記情報記述ファイルを完成させるようにすることもできる。

【0009】この構成によれば、情報記述ファイルの中には、当該ファイル内に含まれる所定の描画命令の数を書き込む必要があるファイル形式において、この描画命令の数を計数した後、再度ファイルに書き込むことをせず、区切り情報領域に記述することができる。したがって、クライアントの、描画命令数を再度書き込む負担を低減することができる。

【0010】さらに、本発明の一つの実施の態様によれば、前記印刷ジョブ作成部は、前後に前記区切り領域を付加された中間形態ファイルを複数個包含する包含ファイルを作成する包含ファイル作成部を含むようにすることもできる。

【0011】さらに、前記包含ファイルを作成する装置においては、前記包含ファイル作成部は、前記複数の中間形態ファイルの各々の格納された領域の位置を示す情報が記述される目次情報領域作成部を含むようにすることもできる。

【0012】さらに、前記包含ファイルを作成する装置においては、前記印刷ジョブ作成部は、先行して前記情報記述ファイルを作成する情報記述ファイル先行作成部を含み、前記区切り領域作成部は、前記先行して作成された情報記述ファイルの区切り情報も作成し、前記包含ファイル作成部は、各々前記区切り領域が付加された、前記先行して作成された情報記述ファイルと前記中間形態ファイルを、包含する包含ファイルを作成するようにすることもできる。

【0013】さらに、前記情報記述ファイルと前記中間形態ファイルの双方を含む包含ファイルを作成する装置においては、前記区切り領域には、当該区切り領域に対応するファイルが、情報記述ファイルであるか中間形態ファイルであるかの情報が記述されるようにすることもできる。

【0014】

【発明の実施の形態】以下、本発明にかかる印刷処理装置の好適な実施の形態（以下、実施形態）を図面に従って説明する。

【0015】図1には、本実施形態の印刷処理装置の概略構成が示されている。ネットワーク10には、クライアント12とプリントサーバ14が接続され、ネットワークを構成している。クライアント12は、図中においては一つ示されているが、複数設けることも可能であり、実際のネットワークとしては複数のクライアントが設けられている例が多数を占める。

【0016】クライアント12には、アプリケーション16からの印刷指示により、ページ記述言語（以下PDLと記す）データを作成するプリンタドライバ18が備

えられている。このPDLは、クライアントごと、またはプリンタごとに複数の種類があり、よって後述するプリントサーバ14は、複数種類のPDLデータを処理可能となっている。プリンタドライバ18は、PDLデータを作成し、これにPDLの種類の情報や出力形式の指示情報などを含むドキュメント・プリンティング・アプリケーション(Document Printing Application)（以下DPAと記す）などのアトリビュートを付加して印刷ジョブを作成する。この印刷ジョブがネットワーク10を介してプリントサーバ14に送信される。

【0017】プリントサーバ14は、前記の印刷ジョブをジョブ受付部20にて受信し、ジョブスケジューラ22によりプリントキュー24への登録を行う。そして、印刷の実行の順番がきた印刷ジョブがスケジューラ22により読み出されコントローラ26に送られる。コントローラ26は、印刷ジョブを受け取ると、これをデコンポーザ・コントロール・ライブラリ28に送出し、DPAアトリビュートに記述されているPDLの種類に応じたデコンポーザ30を選択する。デコンポーザ30は、本プリントサーバ14が印刷を行う可能性のある全てのPDLに対して用意され、各々のPDLデータを解析して、プリンタを駆動するための印刷データに変換される。デコンポーザ30は、たとえば、PDLの一種である、アドビ(Adobe)社のポストスクリプト(PostScript)（商標名）で記述されたデータを変換するためのデコンポーザ30-2や、後述するEMFCを変換するためのEMFCデコンポーザ30-1が備えられている。デコンポーザ・コントロール・ライブラリ28は、デコンポーザ30により変換された印刷データをコントローラ26に送り返し、コントローラ26は印刷データに基づきプリンタ32を駆動し、印刷を実行する。

【0018】本実施形態のクライアント12とプリントサーバ14の間で送受信されるPDLデータには、マイクロソフト(Microsoft)社のウィンドウズ(Windows)（商標名）に規定されたエンハンスド・メタ・ファイル(Enhanced Meta File)（以下EMFと記す）に基づくものがある。EMFは、1ページの印刷情報を一つのファイルに記載する形式であるが、本実施例においては、このEMFを複数個をくるんだ、すなわち複数ページ分の印刷情報を一つのファイルに納めたEMFコンテナ(EMF-Container)（以下EMFCと記す）のファイル形式でPDLデータが作成される。

【0019】図2には、EMFCのファイル形式の構成が示されている。ファイルの先頭にはこれを示すヘッダ100が、またファイルの終端にはこれを示すトレーラ102が設けられている。このヘッダとトレーラによって挟まれた領域が一つの文書（ドキュメント）である。ヘッダ100に続いて、この文書がどのように出力されるべきか、たとえば用紙の大きさ、文字の種類などの情報であるドキュメントアトリビュート104が記述され

ている。

【0020】ドキュメントアトリビュート104に続いてページごとの情報が順次格納されている。一つのページのファイル形態は、先頭にページヘッダ106、続いてページアトリビュート108、ページ内容110、最後にページトレーラ112が記述されている。ページヘッダ106は、ページの大きさに関する情報や後続するページ内容のファイル形式がEMFであるか、後述するモディファイドEMF(modified-EMF)(以下、m-EMFと記す)であるかを示す情報が記述されている。ページアトリビュート108は、後続するページ内容の文字の種類などの情報を示す。ページ内容110は、EMFまたはm-EMFで記述され、ページトレーラ112には、一つのページが終了したことを示す情報が記述される。

【0021】このドキュメントに含まれるページの情報が記述されるとその後にクロスリファレンスが記述される。クロスリファレンス114には、ファイルの先頭からの各ページのヘッダおよびトレーラの位置すなわちオフセットが記述されている。この後にトレーラ102が記述され、ここにはクロスリファレンス114のオフセットが記述されている。

【0022】このように、各ページのオフセットを記述したクロスリファレンス114を設けたことにより、各ページのヘッダ106に直接アクセスでき、さらにページごとのアトリビュート108を有しているので、各ページが独立して扱える。

【0023】次に、m-EMFについて説明する。前述のように、EMFはウインドウズ(商標名)で規定されるPDLであるが、このフォーマットは、ファイル自身に、そのファイルの中で用いた描画処理時に参照される値を記述する必要がある。したがって、ページ内容を順次書き込むときに描画処理時の参照値を計数し、ページ内容の書き込みが全て終了した後この描画処理時の参照値をそのファイルに書き込む必要がある。よって、書き込み動作を2回行う、すなわち2パス動作となる。

【0024】前記の描画処理時に参照される値は、EMFにおいては、フォント、ペン、ブラシの種類などを示すハンドル数と、円、方形などの図形命令、ブラシの色、ペンの太さ・色などを示すレコードの数とがある。そして、たとえば、ある種類のペンを用いる回数がハンドル数であり、このようなハンドル数およびレコード数が一つのEMFファイルについて計数され、これがそのEMFに書き込まれる。

【0025】前述したように、EMFを作成する操作は、2パス処理が必要となる。これは、クライアント12の負担を増加させ、クライアント12における次の作業を行うことができない時間が増加することになる。そこで、本実施形態においては、EMFにおいては、書き戻す必要のあるハンドル数とレコード数をページ内容を

記述する際に計数し、一旦ページトレーラ112に格納する。そして、ページ内容110の中には、ハンドル数とレコード数はダミーの値を記述しておく。このハンドル数とレコード数にダミーの値が記述された、未完成状態のファイルが、本実施形態でいうm-EMFである。

【0026】クライアント12は、このm-EMFの形式でEMFCを作成し、印刷ジョブとして、プリントサーバ14に送出する。プリントサーバ14では、EMFC解析用のEMFCデコンポーザ30-1に印刷ジョブが送られる。EMFCデコンポーザ30-1では、ページトレーラ112に書き込まれたハンドル数とレコード数を読みだし、ページ内容110に書き戻してEMFを完成させる。その後、印刷データに変換する。

【0027】このように、クライアント12で作成されるm-EMFは、EMFに対して中間的な形態であり、この中間形態のファイルが最終的にプリントサーバ14側でEMFとして完成される。したがって、本実施形態によれば、クライアント側の処理は、1パス処理となり負担が軽減される。

【0028】図3および図4には、クライアント12のプリンタドライバ18のEMFCファイルの作成フローが示されている。図3には、基本フローが示されている。まず、アプリケーションからの印刷指示に従いEMFCファイルの作成が開始される(S100)。ファイルが完成するとプリントサーバ14に送信される(S102)。

【0029】図4には、EMFCファイル作成の詳細なフローチャートが示されている。まず、ヘッダが作成され(S104)、続いてドキュメントアトリビュートが作成される(S106)。次に、すでに最終ページに到達したかが判断され(S108)、まだである場合、次のページのアトリビュートが作成される(S110)。次に、ページ内容のm-EMFを作成する(S112)。このとき、ハンドル数、レコード数は0に設定され、見掛け上EMFが作成される。このm-EMFの作成中に、使用したハンドルおよびレコードの数が順次計数され、m-EMF完成時にハンドル数、レコード数が確定する。そして、ページトレーラを作成し、この中にステップS112にて計数したハンドル数、レコード数を書き込み(S114)、ステップS108に戻る。

【0030】ステップS108で最終ページに到達していると判断された場合は、クロスリファレンスが作成され(S116)、続いてトレーラが作成される(S118)。

【0031】図5ないし図7には、プリントサーバ14およびデコンポーザ30の処理フローが示されている。図5には、基本となるフローが示されている。クライアントからの印刷ジョブが受信されると、コントローラ26はデコンポーザ・コントロール・ライブラリ28を経由してデコンポーザ30にゲットプロパティ命令を送信

し、デコンポーザ30はこの命令に対し応答する(S120)。このゲットプロパティ命令は、コントローラ26が、デコンポーザ30がどのような特性(プロパティ)を有しているかを問い合わせる命令であり、たとえばどのようなバージョンに対応できるか、フォントは何をもっているのか、などを問い合わせる。デコンポーザ30は、どのような特性を有しているかを応答する。

【0032】コントローラ26は、ゲットプロパティ命令と同様、ゲットケーパビリティ命令をデコンポーザ30に送信する。ゲットケーパビリティ命令は、コントローラ26が、デコンポーザ30に対してどのような機能(ケーパビリティ)を有しているかを問い合わせる命令であり、たとえば解像度はどのくらいか、用紙サイズはどれに対応可能か、などを問い合わせる。デコンポーザ30は、どのような能力を有しているかを応答する(S122)。

【0033】さらに、コントローラ26は、デコンポーザ30にチェックケーパビリティ命令を送信する。チェックケーパビリティ命令は、実際に所望する機能がデコンポーザ30にあるかを問い合わせる命令である。そして、デコンポーザ30は、指定された機能条件をアンド条件で調べて動作可能であるかをコントローラ26に対し応答する(S124)。

【0034】次に、コントローラ26は、プレスキャン命令をデコンポーザ30に送信し、デコンポーザ30にてEMFCのプレスキャンが行われる(S126)。この詳細は、後述する。プレスキャン終了後、コントローラ26のブレイクページ作成命令により、デコンポーザ30はブレイクページを出力する(S128)。さらに、コントローラ26から指定ページのデコンポーズ命令が送信されると、デコンポーザ30はこの指定ページをデコンポーズする(S130)。指定ページのデコンポーズに関しては後述する。そして、全ての指定ページのデコンポーズが終了したかが判断され(S134)、終了していなければステップS130に戻り、終了していれば、終了処理がなされる。

【0035】図6には、プレスキャンに関するフローチャートが示されている。プレスキャンではまずEMFCのドキュメントアトリビュートが検出される(S136)。次に、トレーラに記述されたクロスリファレンスのオフセット読出し、さらにクロスリファレン스에記述された各ページのオフセットを検出する(S138)。ページオフセットに基づきページ位置を検出し(S140)、各ページアトリビュート(S142)を検出し、プレスキャンが終了する。

【0036】図7には、指定ページのデコンポーズに関するフローチャートが示されている。デコンポーザ30

は、デコンポーズ命令を受けると、EMFCから指定されたページの情報を切り出す(S144)。このとき、予めプレスキャンによりページ位置が検出されているので、直接指定ページを検出することができる。また、プレスキャン時に、ページアトリビュートを検出しているため、このページがm-EMFであるかが分かっている。m-EMFであれば、ページトレーラに記載されたハンドル数、レコード数を検出し、この値をm-EMFに書き込んでEMFを完成させる(S146)。この完成されたEMFをウインドウズ(商標名)のグラフィック・デバイス・インタフェース(GDI)を利用して印刷データにデコンポーズする(S148)。

【0037】以上、本実施形態によれば、PDLとしてEMFを使用する場合においても、クライアント12側で印刷ジョブを作成する際に、1パス処理が可能となる。よって、クライアント12の負担が低減する。

【0038】また、ドキュメントに対するアトリビュートと分離して、各ページごとにアトリビュートを記述しているため、各ページを独立して扱える。

【0039】以上の本実施形態においては、1ページが一つのEMFまたはm-EMFにて作成された場合について説明したが、1ページが複数のEMFまたはm-EMFにて作成されるようにすることもできる。このときには、ページアトリビュートなどに各々のEMFまたはm-EMFをページ内にどのように配置するかを指定するための情報が格納される。さらに、1ページが複数のEMFまたはm-EMF以外のPDL画像または画像データが含まれるようにすることも可能である。

【図面の簡単な説明】

【図1】 本発明にかかる印刷処理装置の実施形態の概略構成図である。

【図2】 本実施形態で用いられるEMFCファイルの形式を示す図である。

【図3】 本実施形態のクライアント側の印刷ジョブに関するフローチャートである。

【図4】 本実施形態のクライアント側のEMFCファイル作成に関するフローチャートである。

【図5】 本実施形態のプリントサーバ側の印刷ジョブに関するフローチャートである。

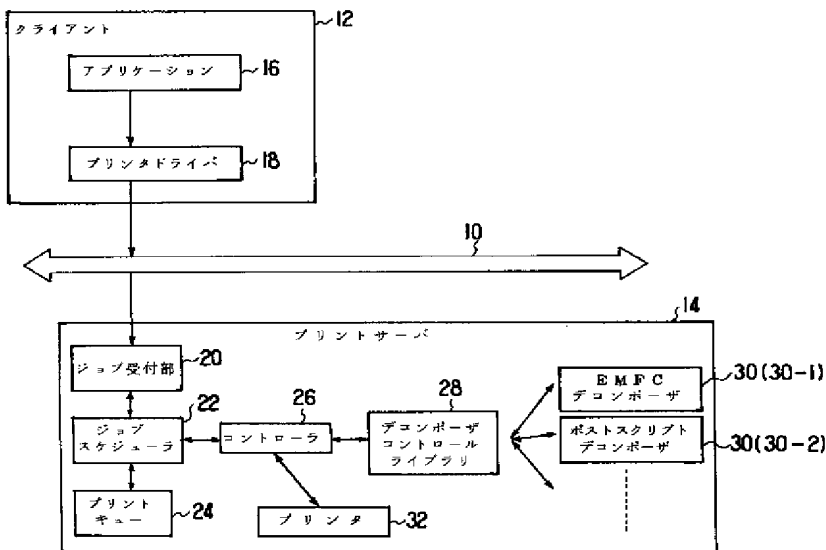
【図6】 本実施形態のプリントサーバ側のプレスキャンに関するフローチャートである。

【図7】 本実施形態のプリントサーバ側のデコンポーズに関するフローチャートである。

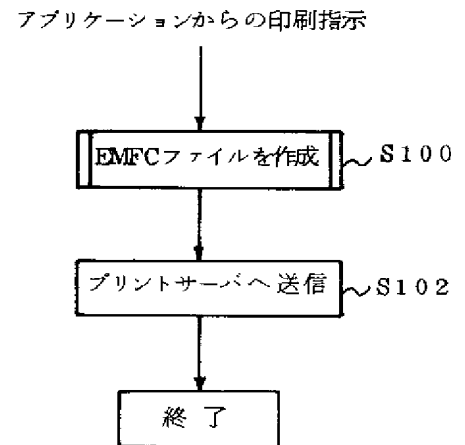
【符号の説明】

10 ネットワーク、12 クライアント、14 プリントサーバ、18 プリンタドライバ、26 コントローラ、30 デコンポーザ。

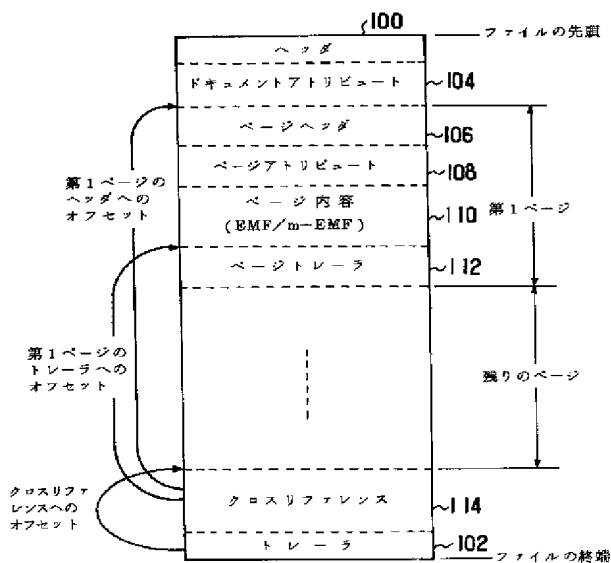
【図1】



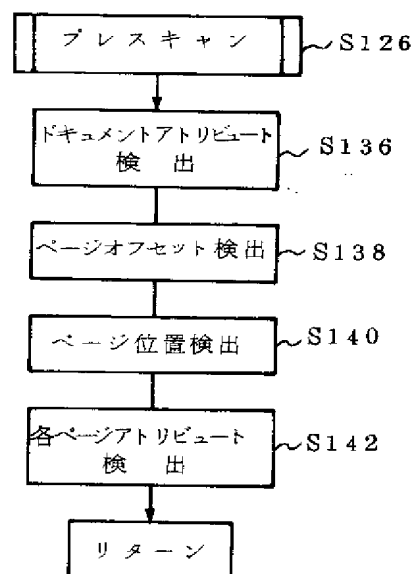
【図3】



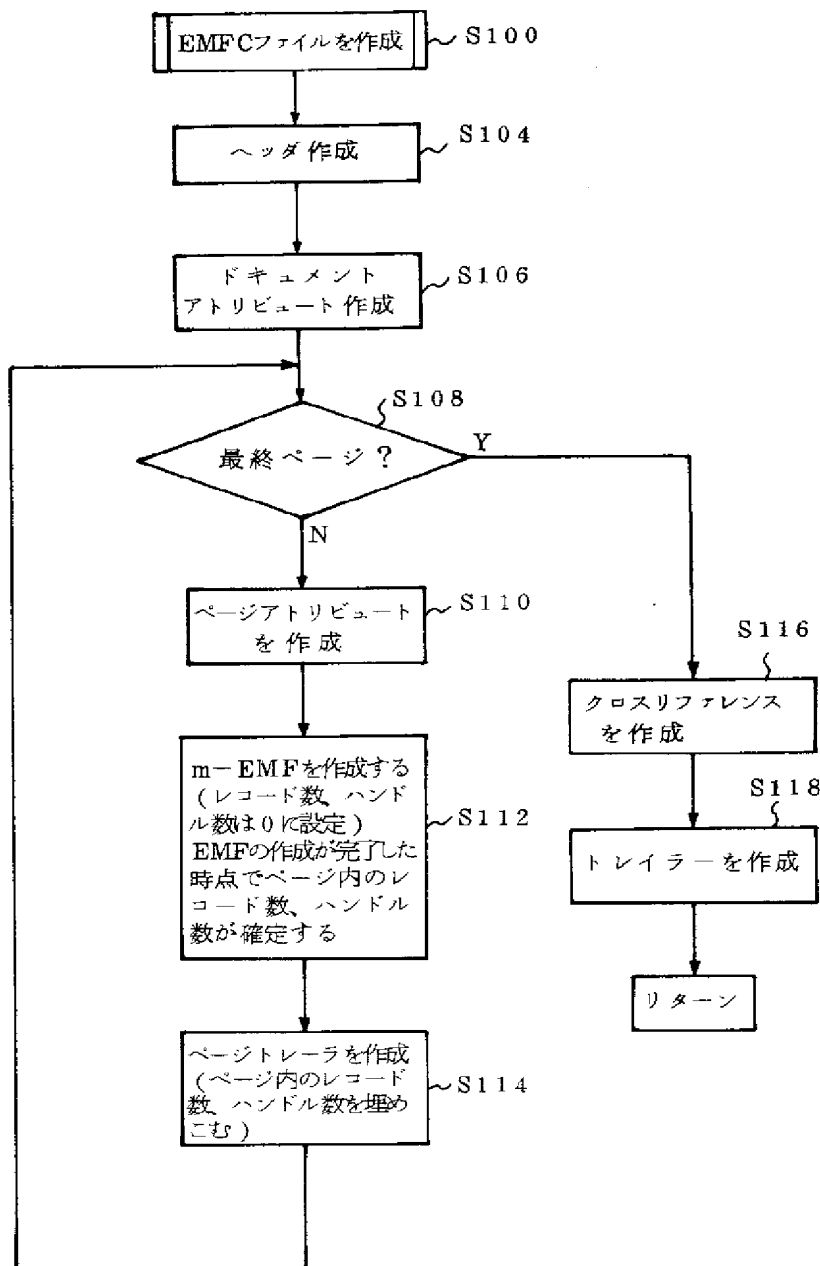
【図2】



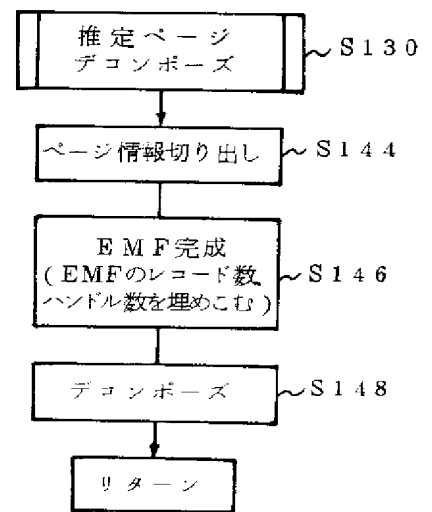
【図6】



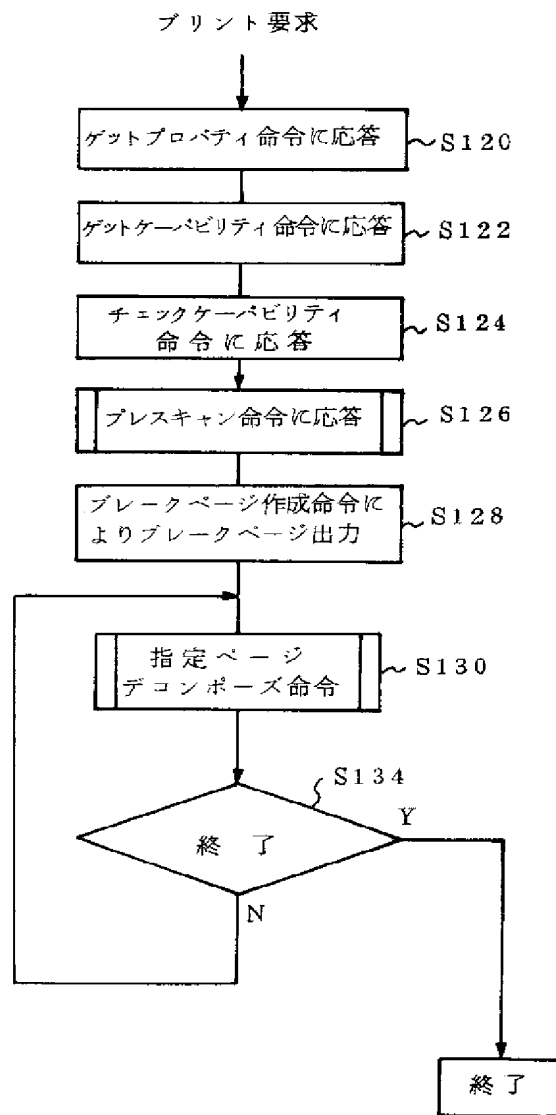
【図4】



【図7】



【図5】



フロントページの続き

(72)発明者 渋谷 俊彦
神奈川県川崎市高津区坂戸3丁目2番1号
KSP R&Dビジネスパークビル 富
士ゼロックス株式会社内

(72)発明者 平松 孝章
神奈川県川崎市高津区坂戸3丁目2番1号
KSP R&Dビジネスパークビル 富
士ゼロックス株式会社内

(72)発明者 江辺 郁夫
神奈川県川崎市高津区坂戸3丁目2番1号
KSP R&Dビジネスパークビル 富
士ゼロックス株式会社内